Language Technology (2023)

Part-of-speech tagging

Marco Kuhlmann
Department of Computer and Information Science

LINKÖPING UNIVERSITY

Parts of speech

- A **part of speech** is a category of words that play similar roles within the syntactic structure of a sentence.
- Three common parts of speech are noun, verb, and adjective.

  Kim loves fast cars.
- There are many different 'tag sets' for parts of speech.

  different languages, different levels of granularity

Universal part-of-speech tags

Source: Universal Dependencies Project

Tag	Category	Examples
ADJ	adjective	*big, old*
ADV	adverb	*very, well*
INTJ	interjection	*ouch!*
NOUN	noun	*girl, cat, tree*
PROPN	proper noun	*Mary, John*
VERB	verb	*run, eat*

Tag	Category	Examples
ADP	adposition	*in, to, during*
AUX	auxiliary verb	*has, should*
CCONJ	conjunction	*and, or, but*
DET	determiner	*a, my, this*
NUM	cardinal numbers	*one, two*
PRON	pronoun	*you, herself*

plus PART, SCONJ, PUNCT, SYM, X

Part-of-speech tagging

- A **part-of-speech tagger** is a computer program that tags each word in a sentence with its part of speech.
- Part-of-speech tagging can be cast as a supervised machine learning problem. This requires training data.

  sentences whose words are tagged with their 'correct' part of speech

Ambiguity causes combinatorial explosion

I	want	to	live	in	peace
PRON	VERB	PART	VERB	ADP	NOUN
NOUN	NOUN	ADP	ADJ	ADV	VERB
		ADV	ADV	ADJ	
				NOUN	

‘I only want to live in peace, plant potatoes, and dream!’ – Moomin

JEOPARDY!

This Stanford University alumnus co-founded educational technology company Coursera.

Source: MacArthur Foundation

SPARQL query against DBPedia

```
SELECT DISTINCT ?x WHERE {
  ?x dbo:almaMater dbr:Stanford_University.
  dbr:Coursera dbo:foundedBy ?x.
}
```

Aspect-based sentiment analysis

NEGATIVE ASPECT

I hated their fajitas,

but their salads were great!

ASPECT POSITIVE

{fajitas: negative, salads: positive}

Pontiki et al. (2014)

Named entity recognition as tagging

- State-of-the algorithms treat named entity recognition as a word-by-word tagging task.

  Just as part-of-speech tagging!

- The basic idea is to use tags that can encode both the boundaries and the types of named entity mentions.

- A common encoding is the **IOB scheme**, where there is a tag for the beginning (B) and inside of each entity type, as well as an additional tag for tokens outside (O) any entity.

Reducing NER to tagging

1	2	3	4	5	6	7	8	9
European	Union	rejects	German	call	to	boycott	British	lamb
B-ORG	I-ORG	O	B-GPE	O	O	O	B-GPE	O
beginning of an ORG	inside an ORG			outside an entity			beginning of a GPE	

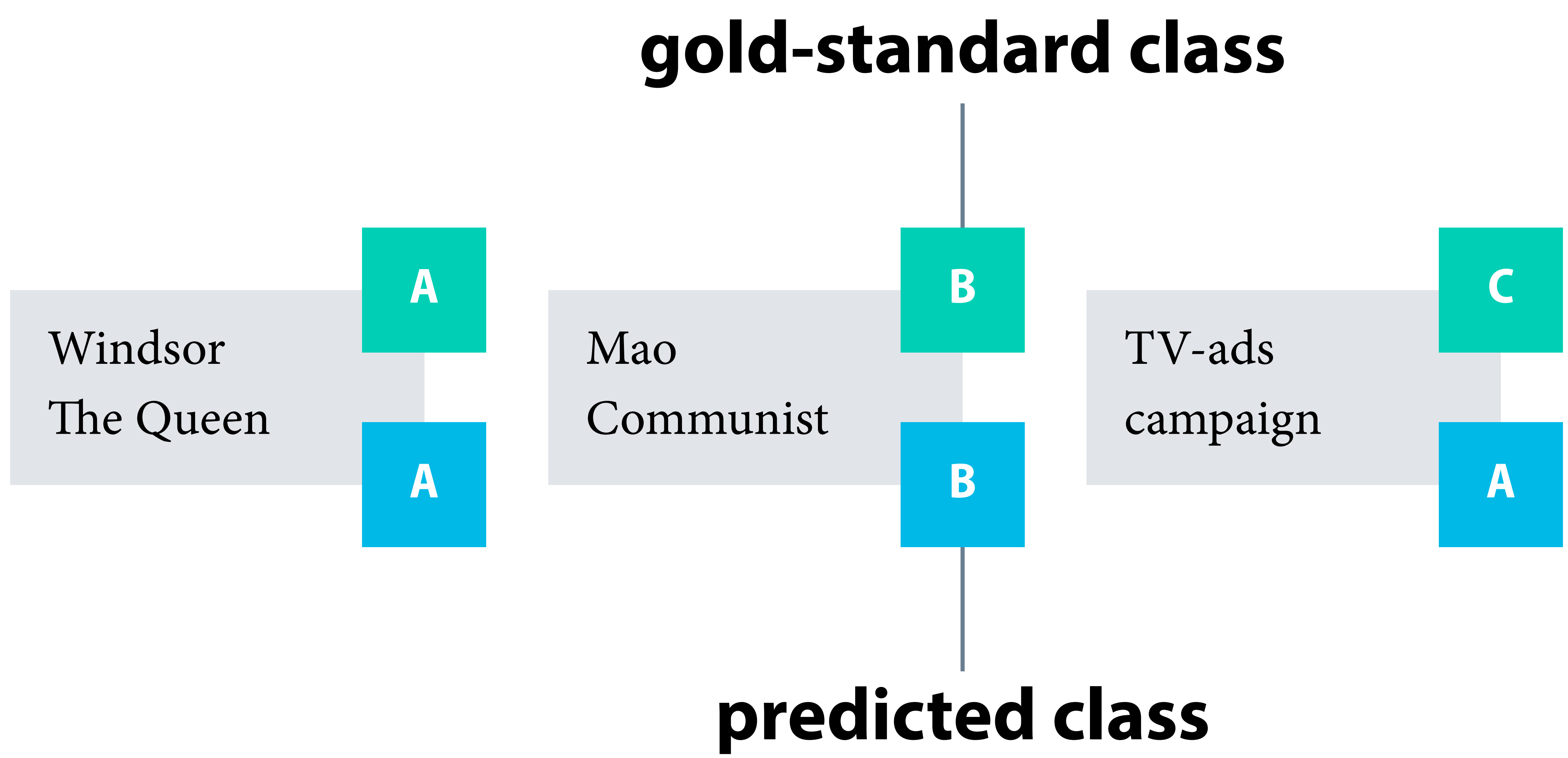
$$\{(1, 2): \textsc{org},\ (4, 4): \textsc{gpe},\ (8, 8): \textsc{gpe}\}$$

This lecture

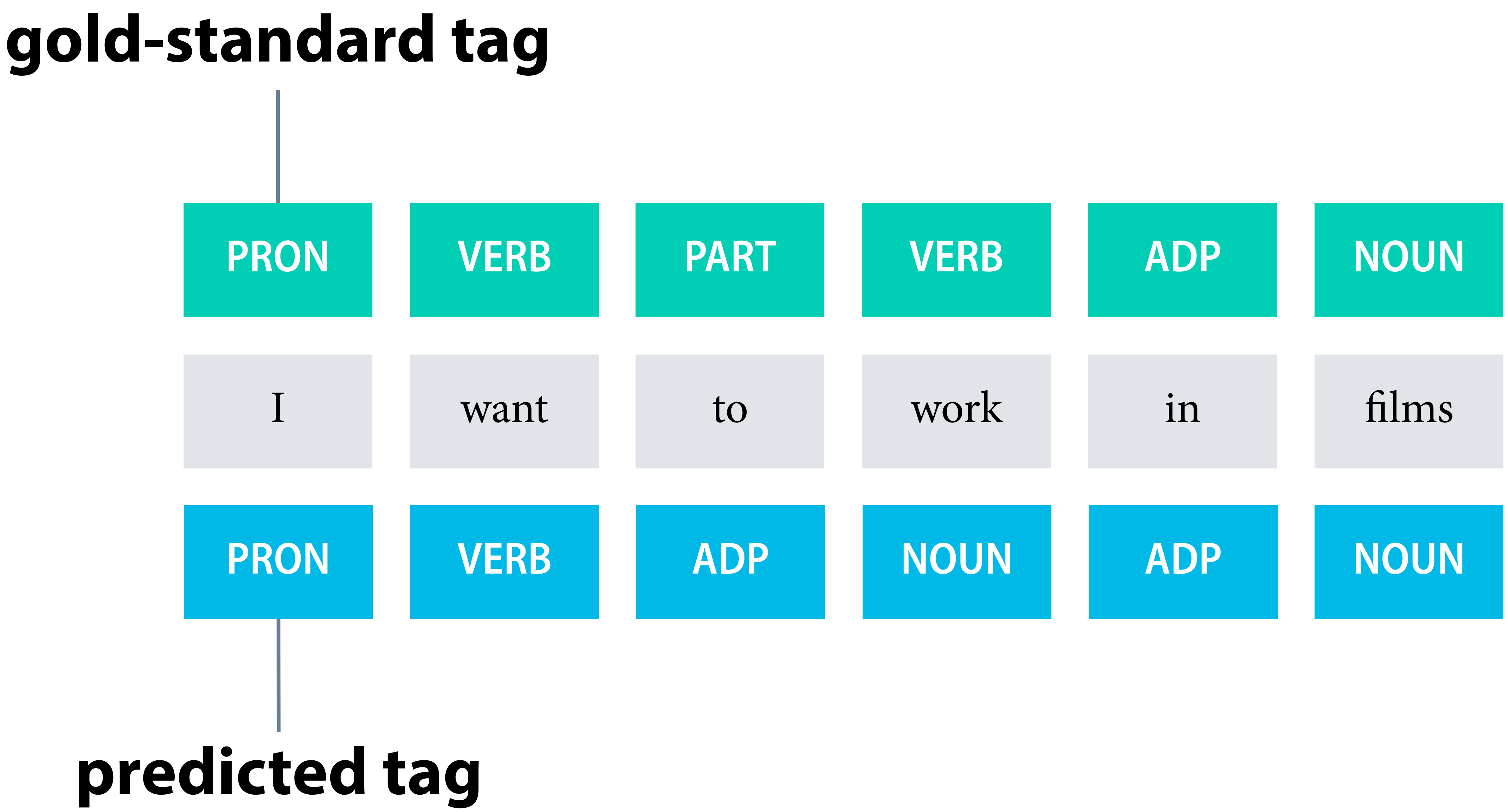
- Introduction to part-of-speech tagging
- Evaluation of part-of-speech taggers
- Part-of-speech tagging with hidden Markov models
- Part-of-speech tagging with multi-class perceptrons

Evaluation of part-of-speech taggers

Reminder: Evaluation of text classifiers



Evaluation of part-of-speech taggers



Stockholm Umeå Corpus (SUC)

- SUC is the largest manually annotated corpus for written Swedish, a collaboration of Stockholm and Umeå University.

  created in the early 1990s

- SUC contains more than 1.1 million tokens; these are annotated with parts of speech, morphological features, and lemmas.

- SUC is a balanced corpus with texts from different genres.

Accuracy

	DET	ADJ	NOUN	ADP	VERB
DET	923	0	0	0	1
ADJ	2	1255	132	1	5
NOUN	0	7	4499	1	18
ADP	0	0	0	2332	1
VERB	0	5	132	2	3436

$$\frac{12445}{12752} = 97.59\%$$

predicted tag

gold-standard tag

Precision with respect to NOUN

	DET	ADJ	NOUN	ADP	VERB
DET	923	0	0	0	1
ADJ	2	1255	132	1	5
NOUN	0	7	4499	1	18
ADP	0	0	0	2332	1
VERB	0	5	132	2	3436

$$\frac{4499}{4763} = 94.46\%$$

predicted tag

gold-standard tag

Recall with respect to NOUN

	DET	ADJ	NOUN	ADP	VERB
DET	923	0	0	0	1
ADJ	2	1255	132	1	5
NOUN	0	7	4499	1	18
ADP	0	0	0	2332	1
VERB	0	5	132	2	3436

$$\frac{4499}{4525} = 99.43\%$$

predicted tag

gold-standard tag

Sample exam question

	NOUN	ADJ	VERB
NOUN	58	6	1
ADJ	5	11	2
VERB	0	7	43

predicted tag

gold-standard tag

Compute (a) precision on adjectives, (b) recall on verbs.

This lecture

- Introduction to part-of-speech tagging
- Evaluation of part-of-speech taggers
- Part-of-speech tagging with hidden Markov models
- Part-of-speech tagging with multi-class perceptrons

Part-of-speech tagging with hidden Markov models

Ambiguity causes combinatorial explosion

I	want	to	live	in	peace
PRON	VERB	PART	VERB	ADP	NOUN
NOUN	NOUN	ADP	ADJ	ADV	VERB
		ADV	ADV	ADJ	
				NOUN	

‘I only want to live in peace, plant potatoes, and dream!’ – Moomin

Relative frequencies of tags per word

I	want	to	live	in	peace
PRON 99.97%	VERB 100.00%	PART 63.46%	VERB 83.87%	ADP 92.92%	NOUN 100.00%
NOUN 0.00%	NOUN 0.00%	ADP 35.13%	ADJ 14.52%	ADV 3.61%	VERB 0.00%
		ADV 0.12%	ADV 0.00%	ADJ 0.03%	
				NOUN 0.27%	

Data: UD English Treebank (training data)

Relative frequencies of next tags per tag

Tag / next tag	ADJ	ADP	ADV	NOUN	PART	PRON	VERB
ADJ	5,22 %	7,93 %	1,34 %	54,70 %	3,26 %	1,37 %	0,94 %
ADP	6,25 %	2,96 %	1,59 %	16,35 %	0,07 %	13,22 %	0,67 %
ADV	13,70 %	8,94 %	10,53 %	1,46 %	1,84 %	8,99 %	19,37 %
NOUN	1,14 %	20,91 %	3,70 %	12,70 %	2,82 %	4,13 %	5,87 %
PART	3,59 %	0,61 %	4,12 %	7,76 %	0,14 %	0,65 %	71,03 %
PRON	3,80 %	3,78 %	5,19 %	13,42 %	1,19 %	2,84 %	27,36 %
VERB	4,32 %	18,13 %	7,25 %	7,72 %	6,74 %	17,01 %	1,62 %

Data: UD English Treebank (training data)

Hidden Markov Model

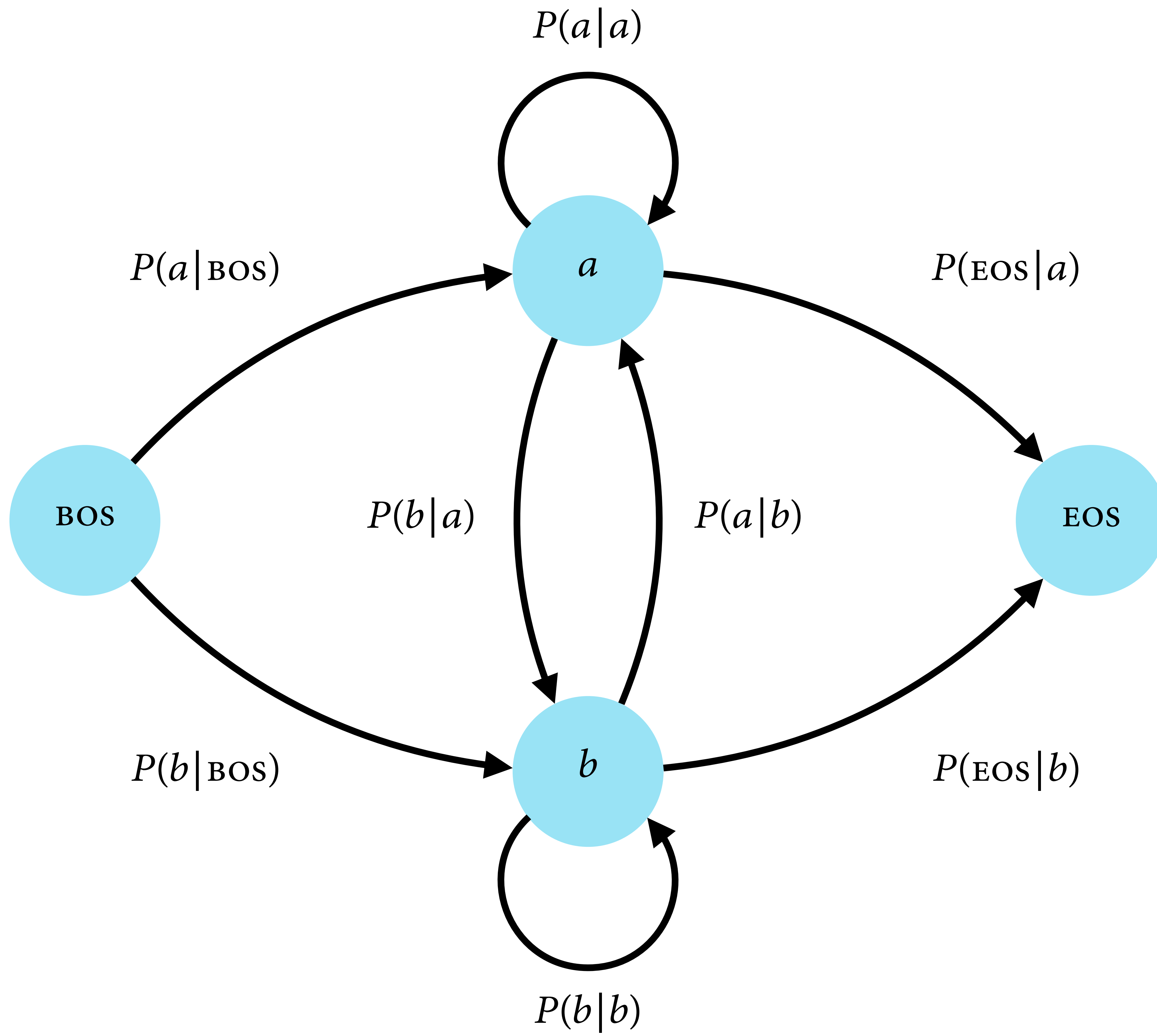
A **hidden Markov model (HMM)** is a generalised Markov model with two types of probabilities:

- transition probabilities $P(\text{next tag} \mid \text{tag})$

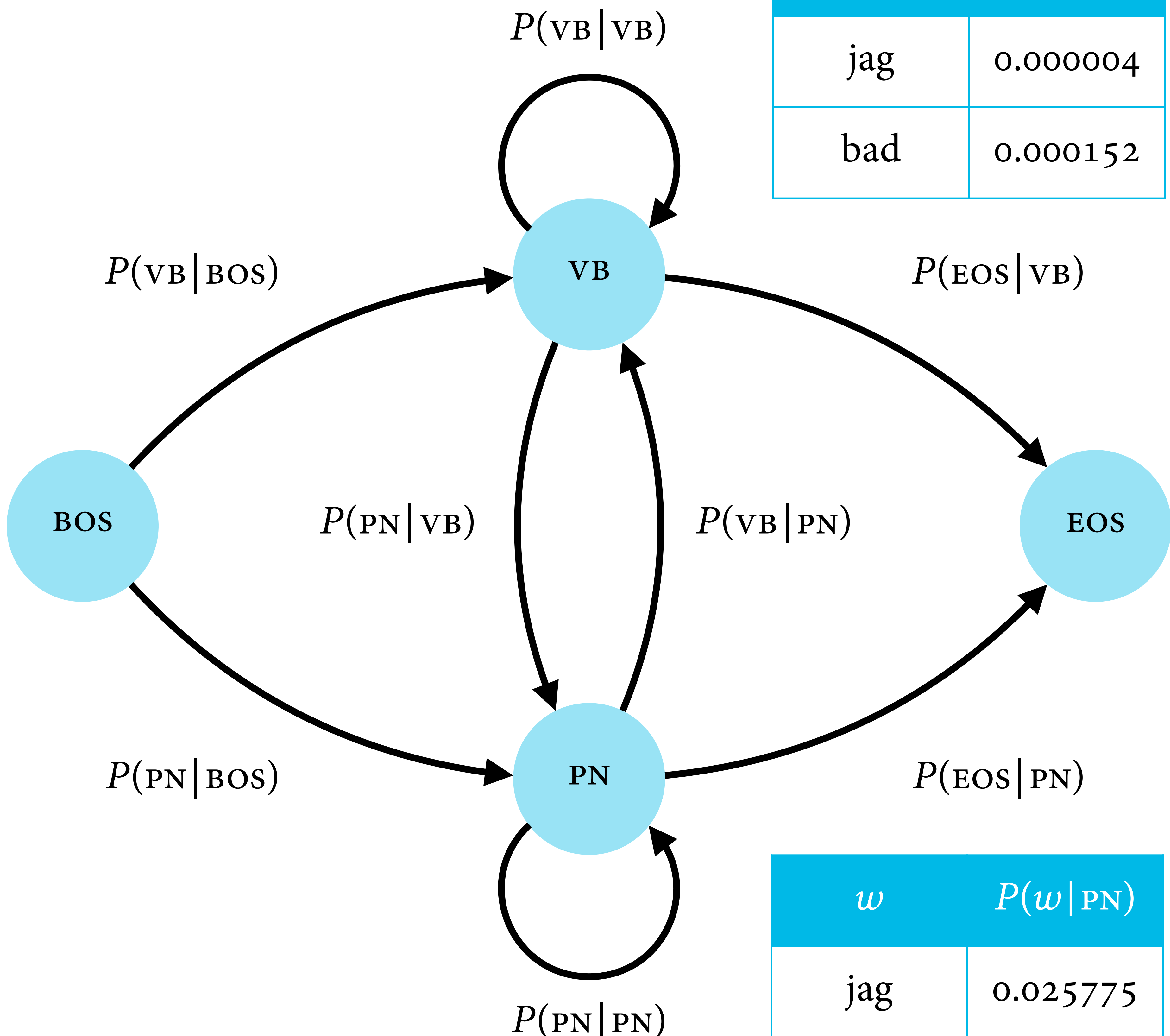
  How probable is it to see a verb after having seen a pronoun?

- output probabilities $P(\text{word} \mid \text{tag})$

  How probable is it to see the word ‘want’ being tagged as a verb?


$P(a|a)$
$P(a|\textsc{bos})$
a
$P(\textsc{eos}|a)$
BOS
$P(b|a)$
$P(a|b)$
EOS
$P(b|\textsc{bos})$
b
$P(\textsc{eos}|b)$
$P(b|b)$

w	$P(w\mid\text{VB})$
jag	0.000004
bad	0.000152



w	$P(w\mid\text{PN})$
jag	0.025775
bad	0.000006

Learning hidden Markov models

To learn a hidden Markov model from a corpus, we can use maximum likelihood estimation just as before:

- To estimate the transition probability $P(\textsc{verb}\,|\,\textsc{pron})$, we ask:

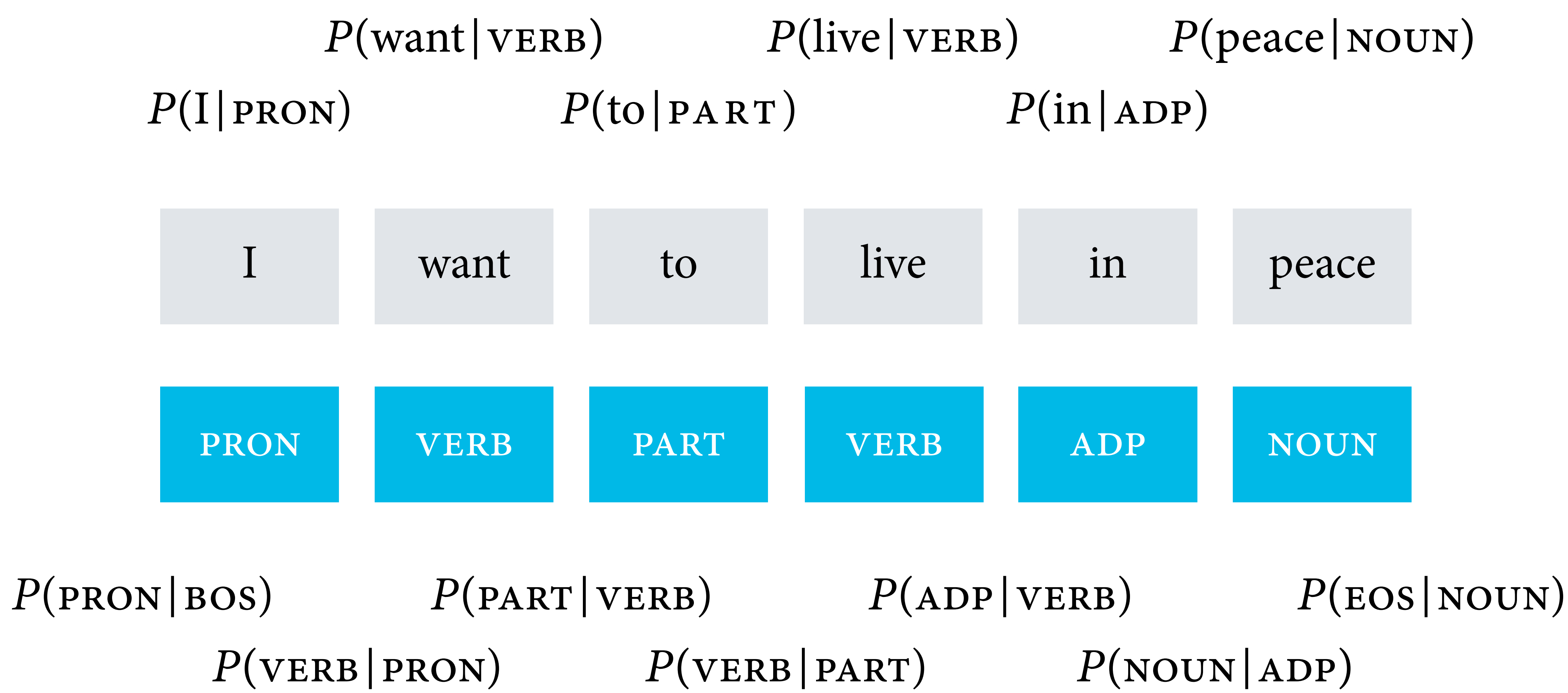
  How often do we see VERB given that the previous tag was PRON?

- To estimate the output probability $P(\text{want}\,|\,\textsc{verb})$, we ask:

  How often do we see the word ‘want’ when the tag is VERB?

We can also use various smoothing techniques just as before.

Probability of a tagged sentence



product of transition and output probabilities

Tagging with a hidden Markov model

- Given a sentence, we want to find a sequence of tags such that the probability of the tagged sentence is maximal.

  The tag sequence is not given in advance; it is 'hidden'!

- For each sentence there are many different tag sequences with many different probabilities.

  combinatorial explosion

- In spite of this, the most probable tag sequence can be found efficiently using the **Viterbi algorithm**.

Sample exam question

You want to compute the probability of this tagged sentence in an HMM:

jag	skrev	på	utan	att	tveka
PN	VB	PL	PP	IE	VB

You can ask the model for its atomic probabilities,
but each such question costs 1 dollar.
How much do you have to pay?

This lecture

- Introduction to part-of-speech tagging
- Evaluation of part-of-speech taggers
- Part-of-speech tagging with hidden Markov models
- Part-of-speech tagging with multi-class perceptrons

Part-of-speech tagging with multi-class perceptrons

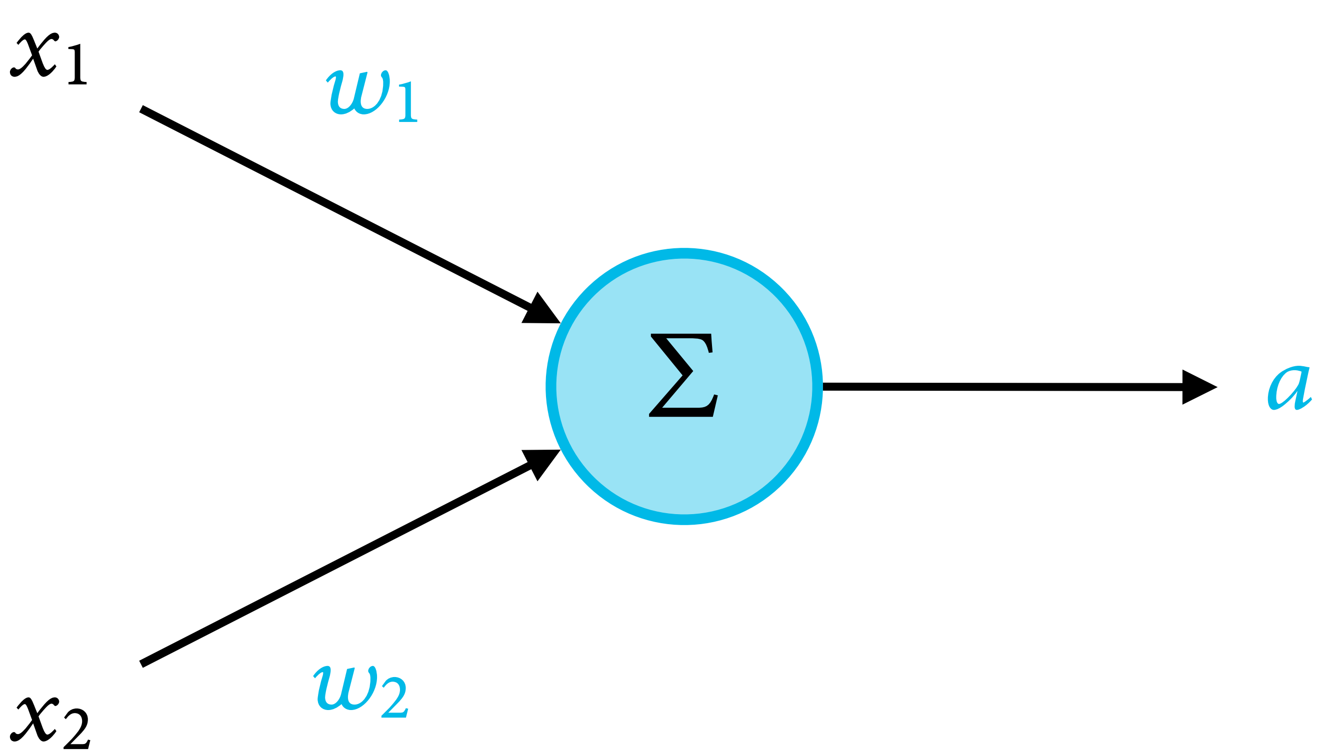
Part-of-speech tagging as classification

- Part-of-speech tagging can be cast as a sequence of classification problems – one classification per word in the sentence.
- Based on this idea, any method for classification can be used to build a part-of-speech tagger.

  Naive Bayes

- Here we use a very simple non-probabilistic method called the **multi-class perceptron**.

The classical perceptron



$$f(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{xw} > 0 \\ 0 & \text{otherwise} \end{cases}$$

linear model + decision rule (threshold)

Inspiration from neurobiology

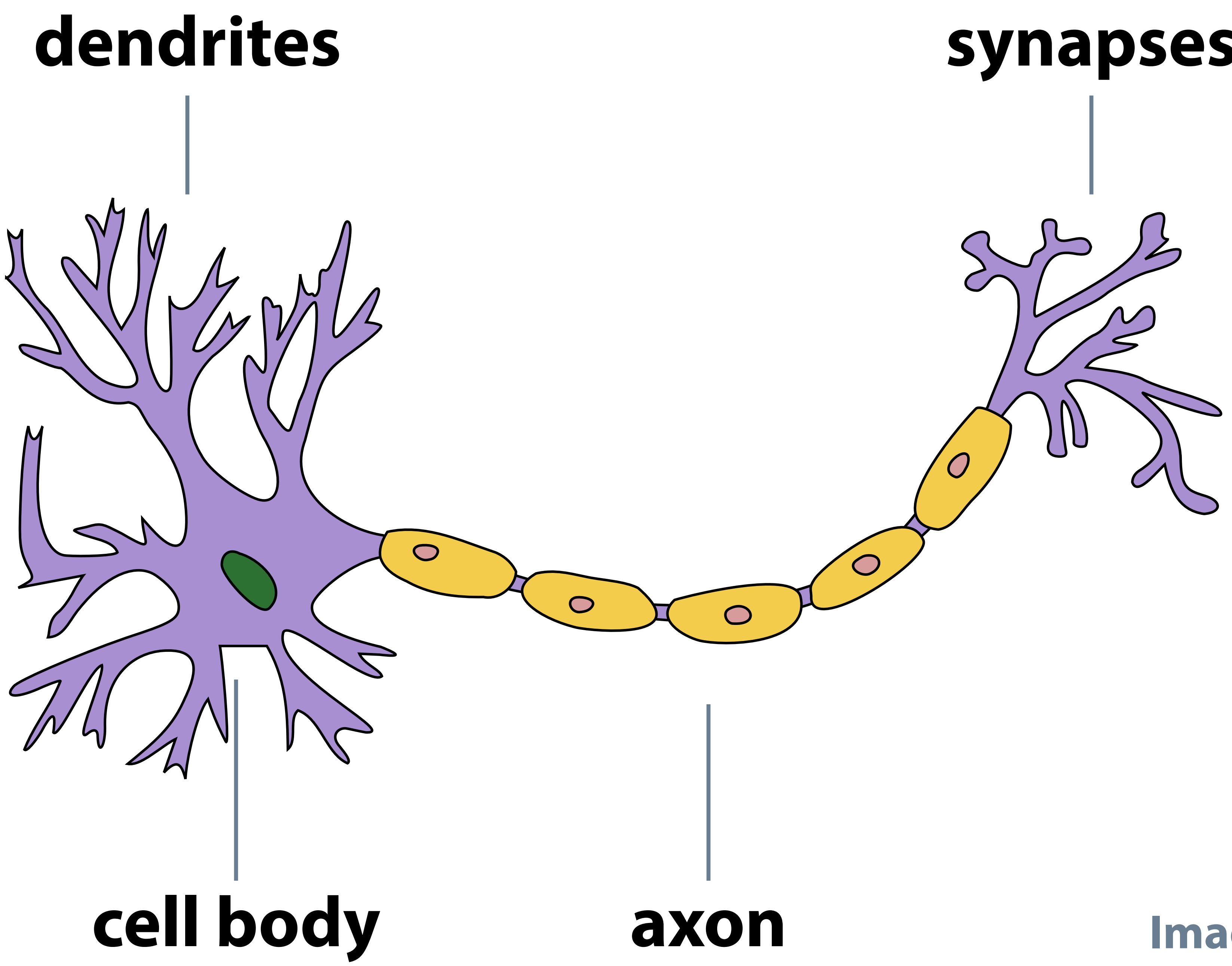
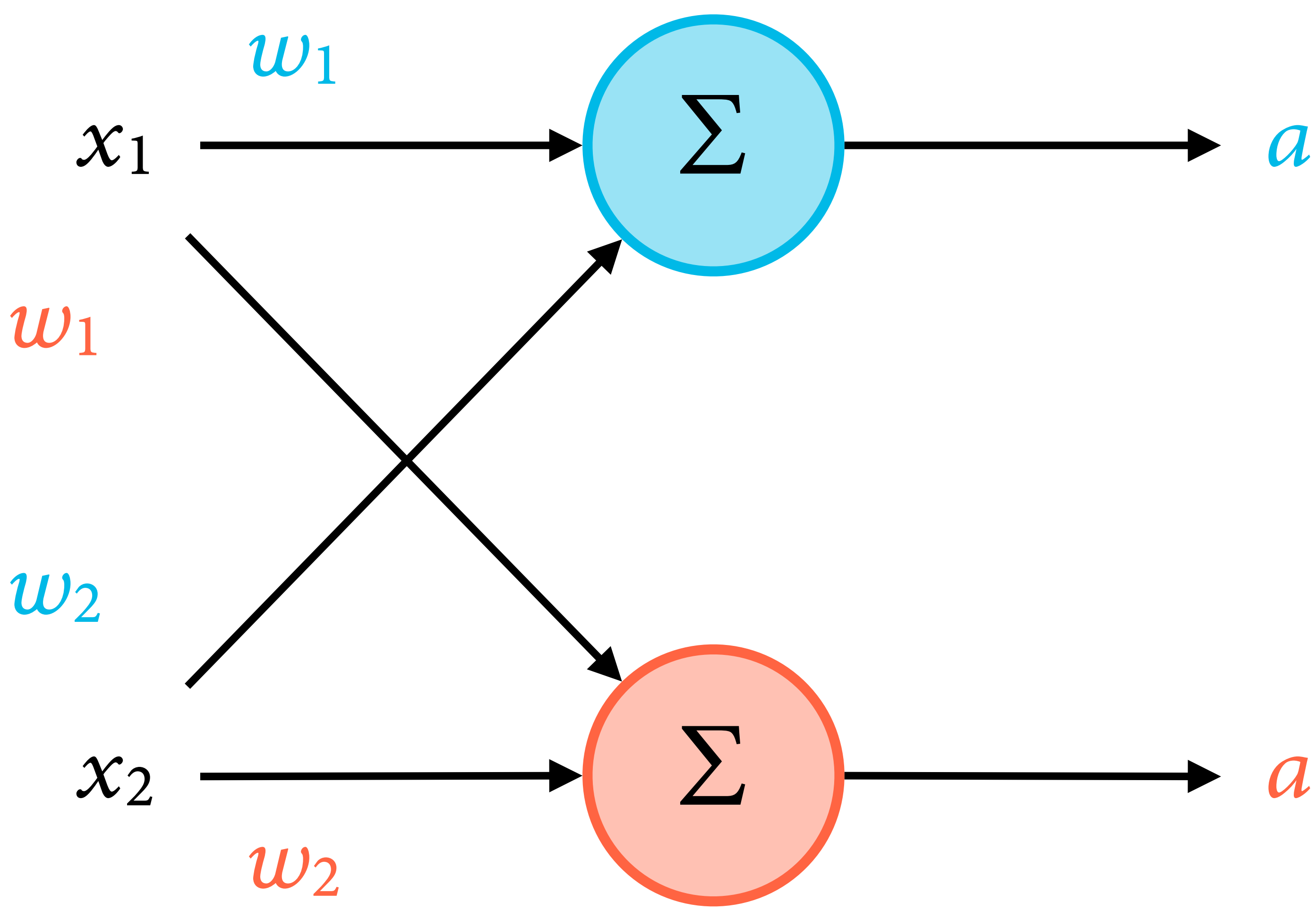


Image source: Wikipedia

The multi-class perceptron



$$f(\boldsymbol{x}) = \underset{c}{\operatorname{argmax}}\ \boldsymbol{x}\boldsymbol{w}_c$$

linear model + decision rule (argmax)

Feature vectors

- In order to use the perceptron to classify data, we need to represent data samples as vectors.

  Slogan: We ‘featurize’ the data.

- Intuitively, the **feature vector** for a sample determines how the perceptron ‘sees’ this sample of the data.

- For most of the discussion here we will be assuming feature vectors whose values are non-negative floats.

Weight vectors

- Features whose weights are zero do not contribute to the activation; such features are ignored.
- Features whose weights are positive cause the activation to increase – they suggest that $\boldsymbol{x}$ *does belong* to the class at hand.
- Features whose weights are negative cause the activation to decrease – they suggest that $\boldsymbol{x}$ *does not belong* to the class.

This assumes that feature values are non-negative floats.

Part-of-speech tagging with a perceptron

I	want	to	live	in	peace
NOUN	9.36				
PRON	81.72				
VERB	−9.18				

Part-of-speech tagging with a perceptron

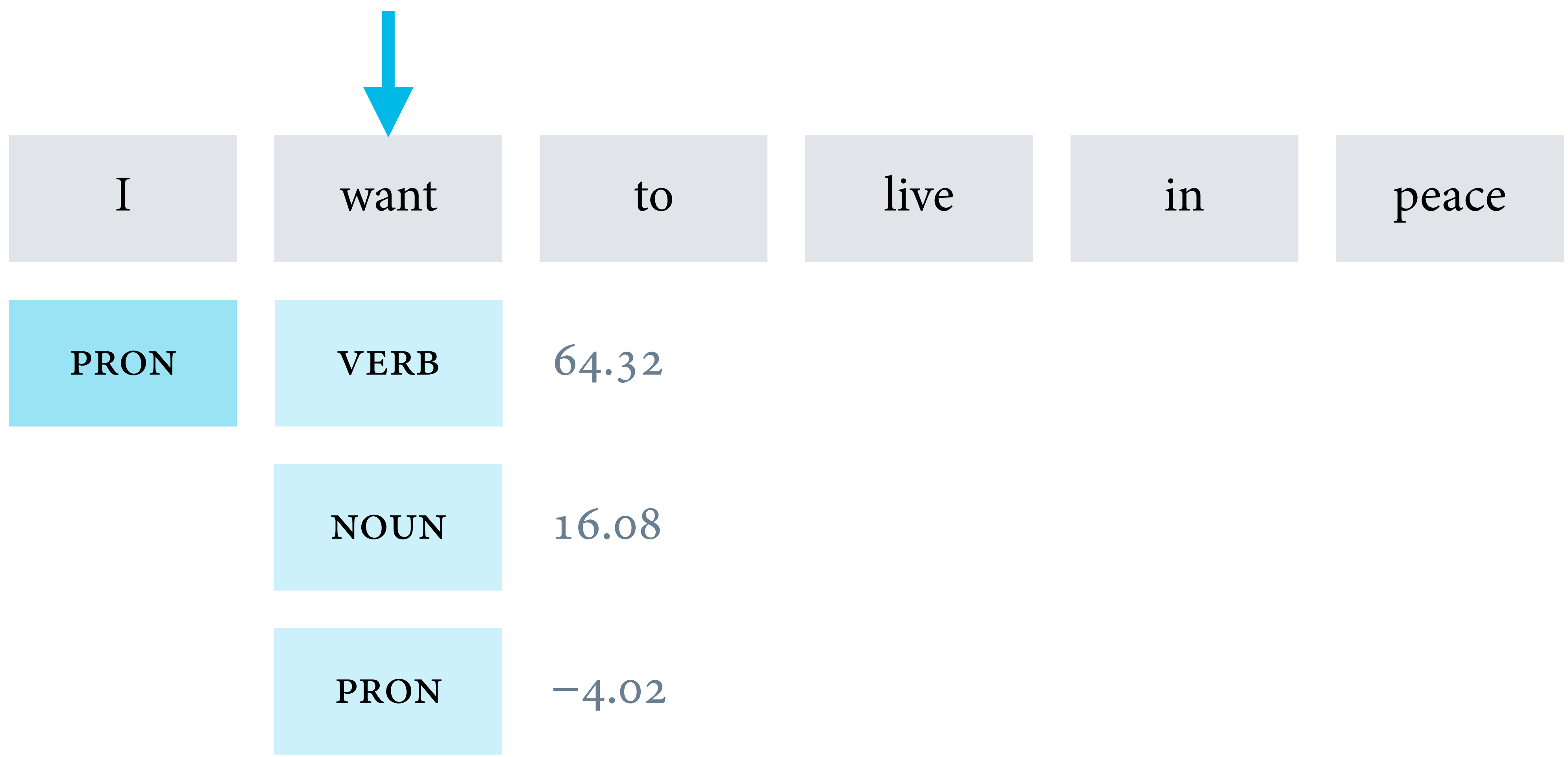
I	want	to	live	in	peace

PRON	81.72
NOUN	9.36
VERB	−9.18

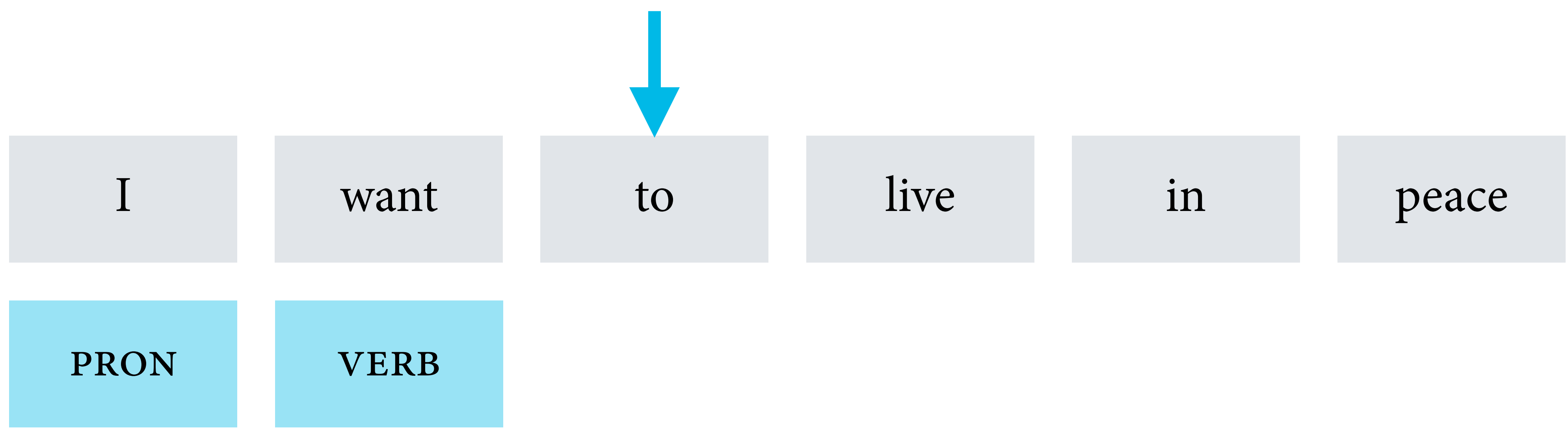
Part-of-speech tagging with a perceptron

I	want	to	live	in	peace	
PRON	NOUN	16.08				
	PRON	−4.02				
	VERB	64.32				

Part-of-speech tagging with a perceptron



Part-of-speech tagging with a perceptron



Feature windows

- Hidden Markov models look back one step; but sometimes it is a good idea to look back further, or to look ahead!

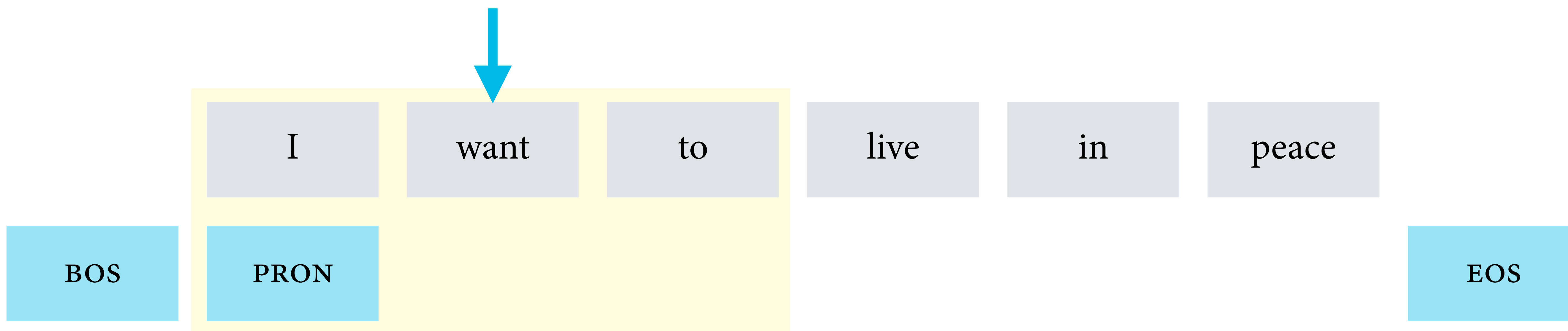
  I want <u>to</u> live in peace.

- At the same time, we do not want the classifier to 'see' too much information.

  efficiency, data sparseness

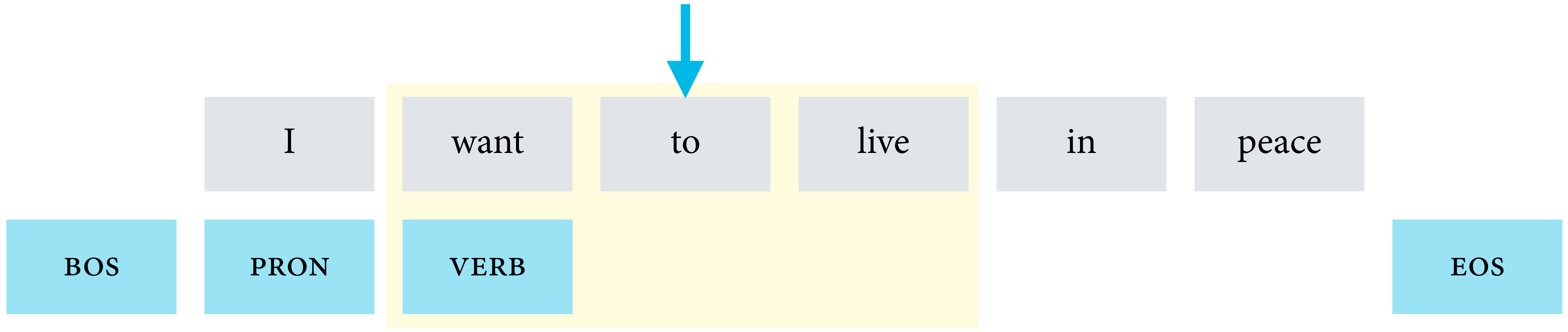
- A compromise is to define a limited **feature window**.

Feature window



With this feature window, we 'see' the current word, the previous word, the next word, and the previous tag.

Feature window



The feature window moves forward during tagging.

Examples of features in part-of-speech tagging

- (lowercase) word form of the current token
- word forms of the preceding tokens, next tokens
- capitalisation of the current token (upper, lower, N/A)
- type of the current token (digits, letters, symbols)
- various prefixes and suffixes of the current token
- whether the current token is hyphenated
- whether the token is first or last in the sentence
- various combinations of the features above

Östling (2013)

Comparison between the two methods

Part-of-speech tagging with hidden Markov models

- probabilistic
- exhaustive search for the best sequence (Viterbi algorithm)
- limited possibilities to define features (current word, previous tag)

Part-of-speech tagging with multi-class perceptrons

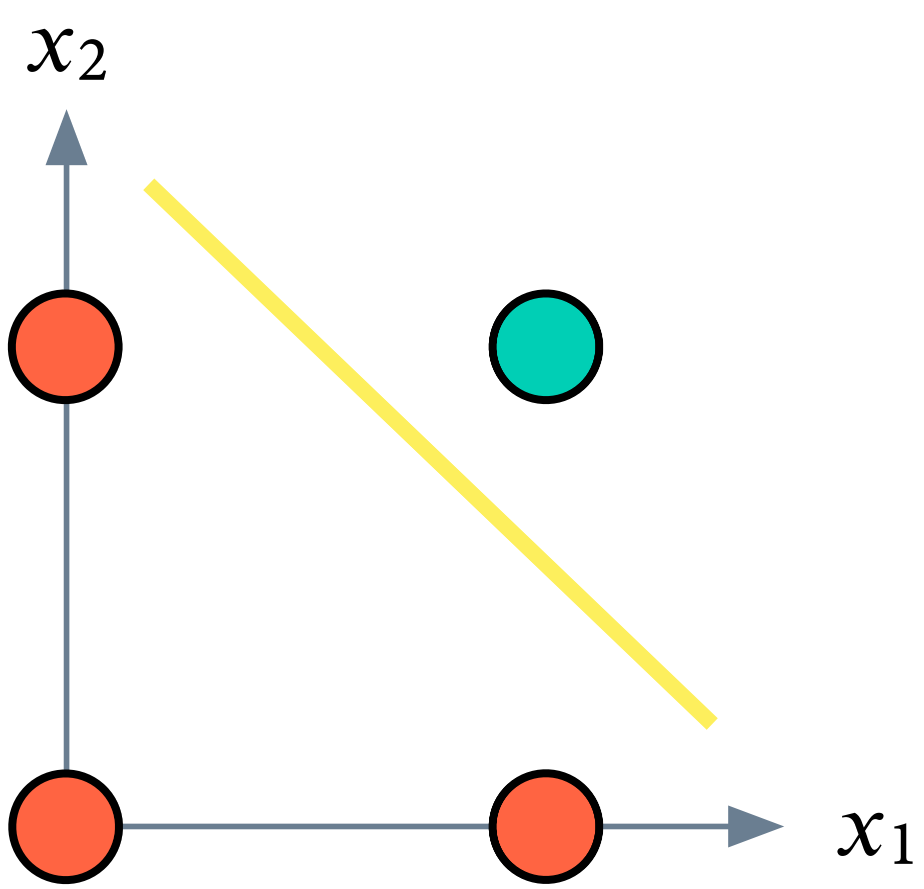
- non-probabilistic
- no search; locally optimal decisions
- more possibilities to define features (feature windows)

Comparison between the two methods

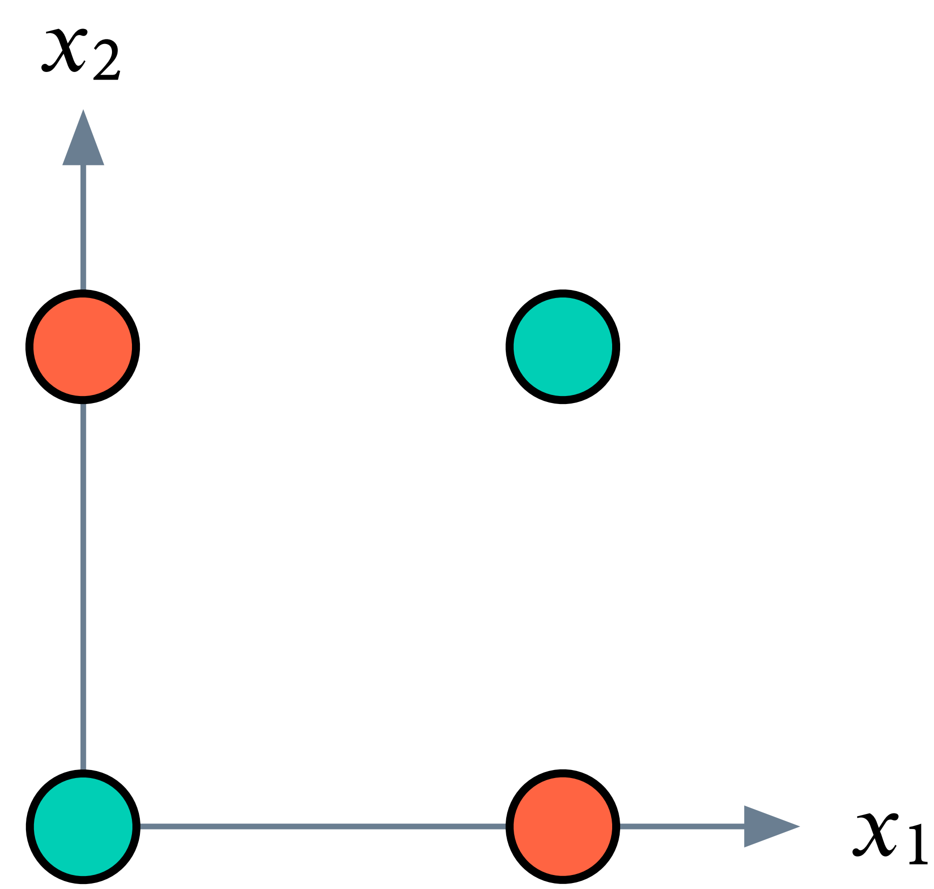
Hidden markov model		Multi-class perceptron	
Viterbi search	greedy search	HMM features	fine-tuned features
92.71 %	89.97 %	88.86 %	95.30 %

Tagging accuracy on the SUC test set

Limitations of the perceptron

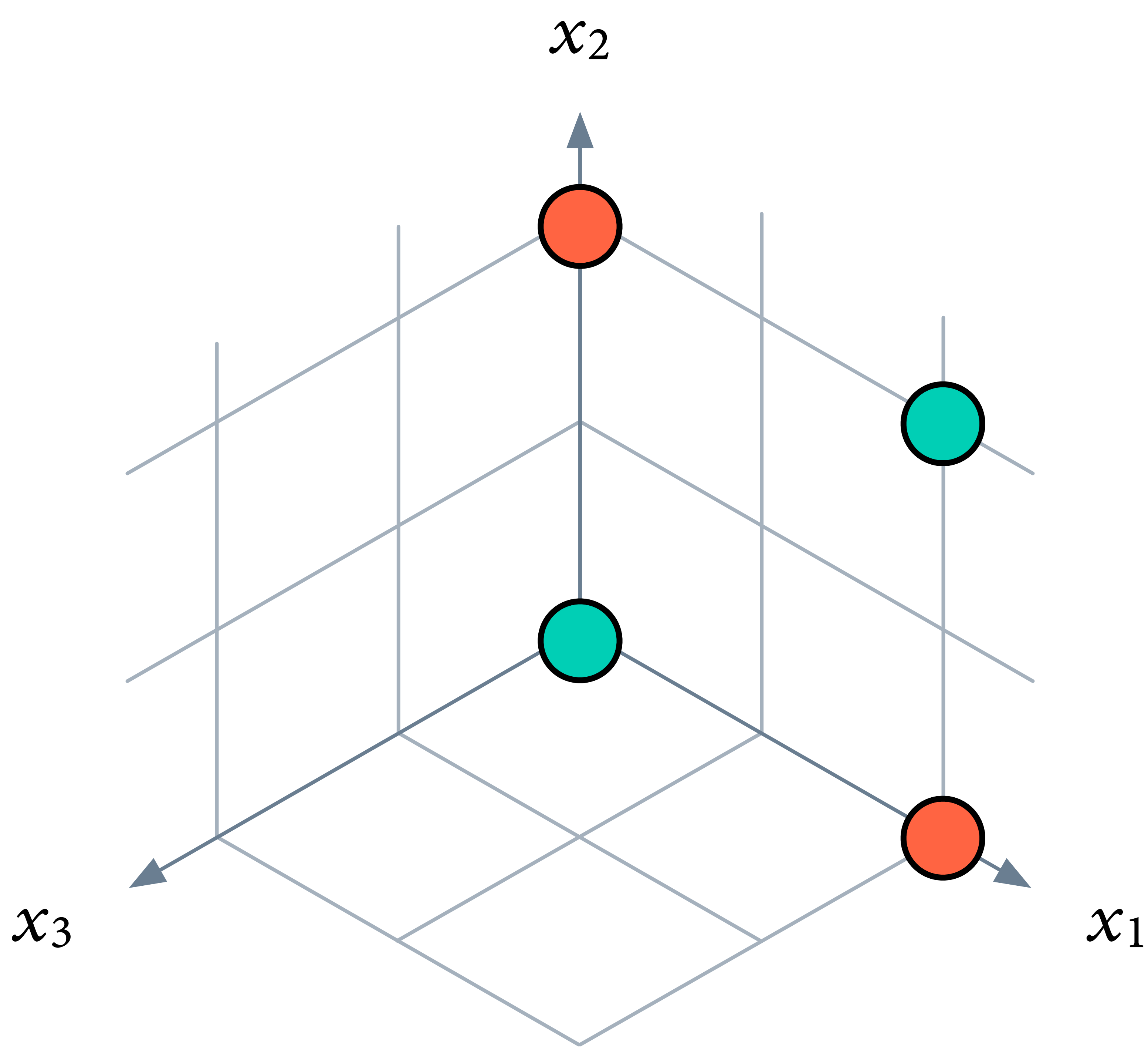


linearly separable

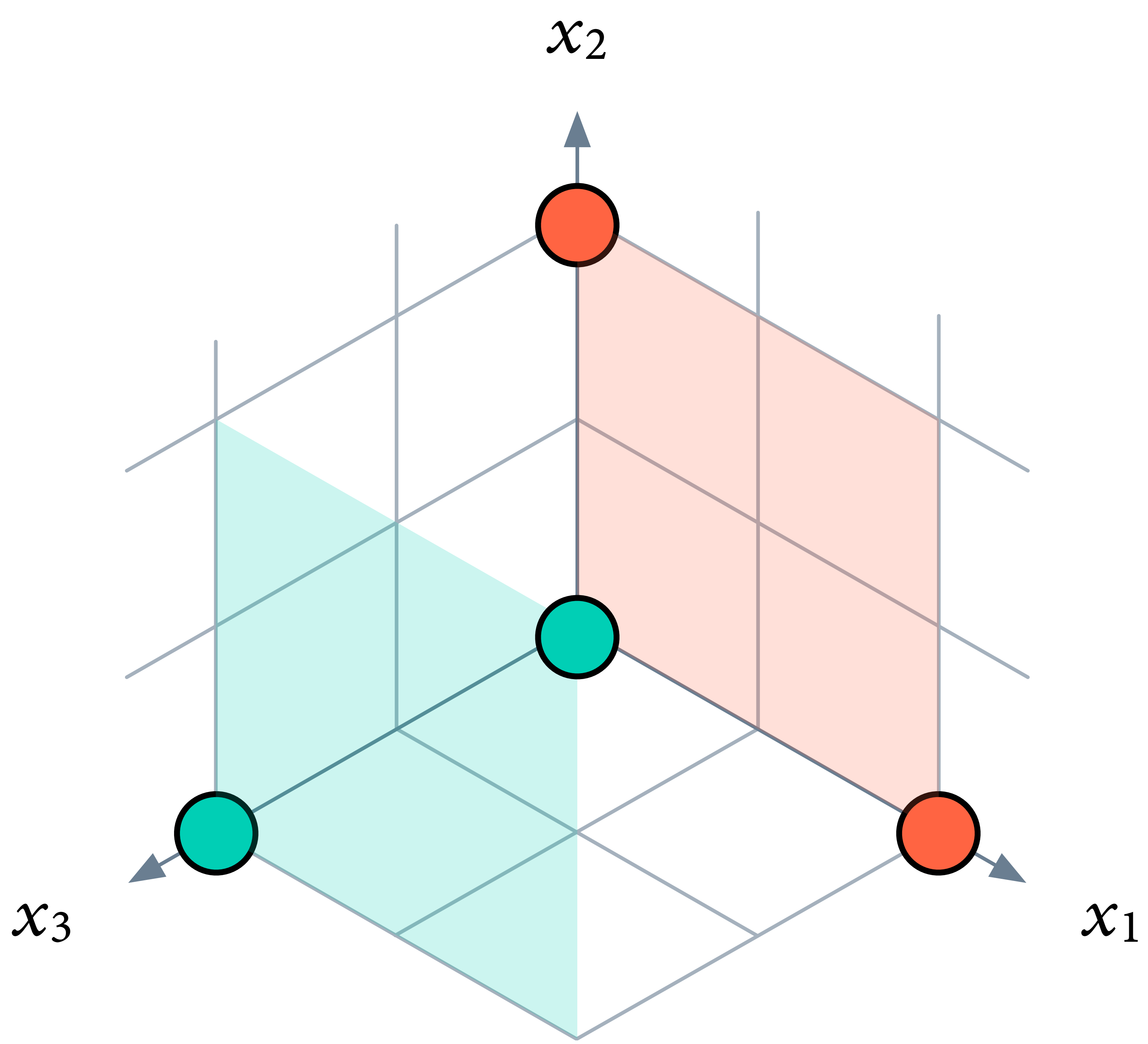


not linearly separable

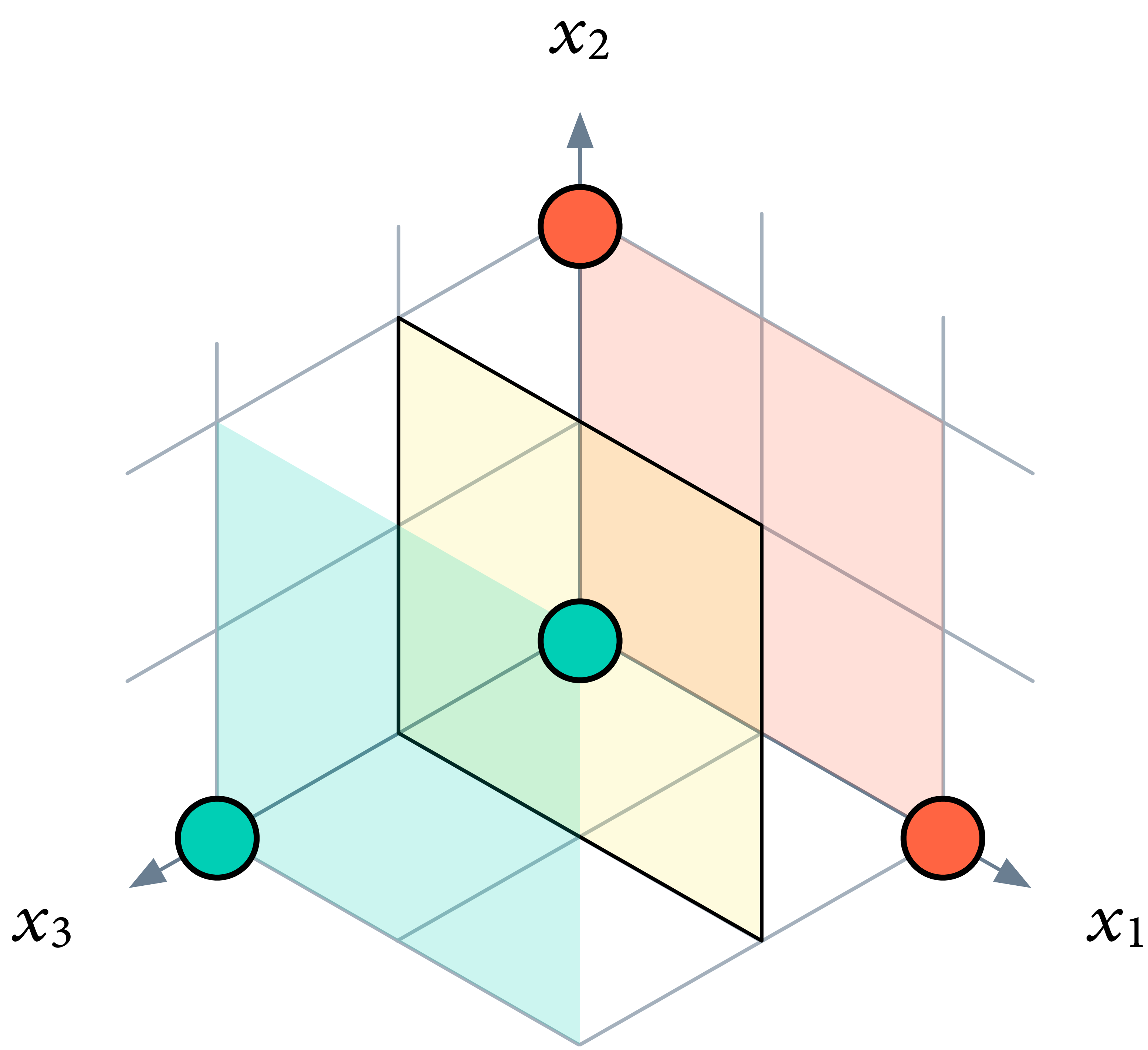
New features to the rescue!



New features to the rescue!



New features to the rescue!



How do we get new features?

Suppose that we could apply the linear model not to $\boldsymbol{x}$ directly but to a representation $\phi(\boldsymbol{x})$ of $\boldsymbol{x}$. How could we get this representation?

- **Option 1.** Manually engineer ϕ using expert knowledge.

  feature engineering – linear classifiers

- **Option 2.** Make the model sensitive to parameters such that learning these parameters identifies a good representation ϕ.

  feature learning – neural networks

This lecture

- Introduction to part-of-speech tagging
- Evaluation of part-of-speech taggers
- Part-of-speech tagging with hidden Markov models
- Part-of-speech tagging with multi-class perceptrons